

# Strategy Improvement for Concurrent Safety Games\*

Krishnendu Chatterjee<sup>§</sup>      Luca de Alfaro<sup>§</sup>      Thomas A. Henzinger<sup>†‡</sup>

<sup>§</sup> CE, University of California, Santa Cruz, USA

<sup>†</sup> EECS, University of California, Berkeley, USA

<sup>‡</sup> Computer and Communication Sciences, EPFL, Switzerland  
{c\_krish,tah}@eecs.berkeley.edu, luca@soe.ucsc.edu

## Abstract

We consider concurrent games played on graphs. At every round of the game, each player simultaneously and independently selects a move; the moves jointly determine the transition to a successor state. Two basic objectives are the safety objective: “stay forever in a set  $F$  of states”, and its dual, the reachability objective, “reach a set  $R$  of states”. We present in this paper a strategy improvement algorithm for computing the *value* of a concurrent safety game, that is, the maximal probability with which player 1 can enforce the safety objective. The algorithm yields a sequence of player-1 strategies which ensure probabilities of winning that converge monotonically to the value of the safety game.

The significance of the result is twofold. First, while strategy improvement algorithms were known for Markov decision processes and turn-based games, as well as for concurrent reachability games, this is the first strategy improvement algorithm for concurrent safety games. Second, and most importantly, the improvement algorithm provides a way to approximate the value of a concurrent safety game *from below* (the known value-iteration algorithms approximate the value from above). Thus, when used together with value-iteration algorithms, or with strategy improvement algorithms for reachability games, our algorithm leads to the first practical algorithm for computing converging upper and lower bounds for the value of reachability and safety games.

## 1 Introduction

We consider games played between two players on graphs. At every round of the game, each of the two players selects a move; the moves of the players then determine the transition to the successor state. A play of the game gives rise to a path on the graph. We consider two basic goals for the players: *reachability*, and *safety*. In the reachability goal, player 1 must reach a set of target states or, if randomization is needed to play the game, then player 1 must maximize the probability of reaching the target set. In the safety goal, player 1 must ensure that a set of target states is never left or, if randomization is required, then player 1 must ensure that the probability of leaving the target set is as low as possible. The two goals are dual, and the games are determined: the maximal probability with which player 1 can reach a target set is equal to one minus the maximal probability with which player 2 can confine the game in the complement set [18].

---

\*This research was supported in part by the NSF grants CCR-0132780, CNS-0720884, and CCR-0225610, and by the Swiss National Science Foundation.

These games on graphs can be divided into two classes: *turn-based* and *concurrent*. In turn-based games, only one player has a choice of moves at each state; in concurrent games, at each state both players choose a move, simultaneously and independently, from a set of available moves.

For turn-based games, the solution of games with reachability and safety goals has long been known. If the move played determines uniquely the successor state, the games can be solved in linear-time in the size of the game graph. If the move played determines a probability distribution over the successor state, the problem of deciding whether a safety or reachability can be won with probability greater than  $p \in [0, 1]$  is in  $\text{NP} \cap \text{co-NP}$  [5], and the exact value of a game can be computed by strategy improvement algorithms [6]. These results all hinge on the fact that turn-based reachability and safety games can be optimally won with deterministic, and memoryless, strategies. These strategies are functions from states to moves, so they are finite in number, and guarantees the termination of the algorithms.

The situation is different for the concurrent case, where randomization is needed even in the case in which the moves played by the players uniquely determine the successor state. The *value* of the game is defined, as usual, as the sup-inf value: the supremum, over all strategies of player 1, of the infimum, over all strategies of player 2, of the probability of achieving the safety or reachability goal. In concurrent reachability games, players are only guaranteed the existence of  $\varepsilon$ -optimal strategies, that ensure that the value of the game is achieved within a specified  $\varepsilon > 0$  [17]; these strategies (which depend on  $\varepsilon$ ) are memoryless, but in general need randomization [10]. However, for concurrent safety games memoryless optimal strategies exist [11]. Thus, these strategies are mappings from states, to probability distributions over moves.

While complexity results are available for the solution of concurrent reachability and safety games, practical algorithms for their solution, that can provide both a value, and an estimated error, have so far been lacking. The question of whether the value of a concurrent reachability or safety game is at least  $p \in [0, 1]$  can be decided in PSPACE via a reduction to the theory of the real closed field [13]. This yields a binary-search algorithm to approximate the value. This approach is theoretical, but complex due to the complex decision algorithms for the theory of reals.

Thus far, the only practical approach to the solution of concurrent safety and reachability games has been via value iteration, and via strategy improvement for reachability games. In [11] it was shown how to construct a series of valuations that approximates from below, and converges, to the value of a reachability game; the same algorithm provides valuations converging from above to the value of a safety game. In [4], it was shown how to construct a series of strategies for reachability games that converge towards optimality. Neither scheme is guaranteed to terminate, not even strategy improvement, since in general only  $\varepsilon$ -optimal strategies are guaranteed to exist. Both of these approximation schemes lead to practical algorithms. The problem with both schemes, however, is that they provide only *lower* bounds for the value of reachability games, and only *upper* bounds for the value of safety games. As no bounds are available for the speed of convergence of these algorithms, the question of how to derive the matching bounds has so far been open.

In this paper, we present the first strategy improvement algorithm for the solution of concurrent safety games. Given a safety goal for player 1, the algorithm computes a sequence of memoryless, randomized strategies  $\pi_1^0, \pi_1^1, \pi_1^2, \dots$  for player 1 that converge towards optimality. Albeit memoryless randomized optimal strategies exist for safety goals [11], the strategy improvement algorithm may not converge in finitely many iterations: indeed, optimal strategies may require moves to be played with irrational probabilities, while the strategies produced by the algorithm play moves with probabilities that are rational numbers. The main significance of the algorithm is that it provides

a converging sequence of *lower* bounds for the value of a safety game, and dually, of *upper* bounds for the value of a reachability game. To obtain such bounds, it suffices to compute the value  $v_k(s)$  provided by  $\pi_1^k$  at a state  $s$ , for  $k > 0$ . Once  $\pi_1^k$  is fixed, the game is reduced to a Markov decision process, and the value  $v_k(s)$  of the safety game can be computed at all  $s$  e.g. via linear programming [7, 3]. Thus, together with the value or strategy improvement algorithms of [11, 4], the algorithm presented in this paper provides the first practical way of computing converging lower and upper bounds for the values of concurrent reachability and safety games. We also present a detailed analysis of termination criteria for turn-based stochastic games, and obtain an improved upper bound for termination for turn-based stochastic games.

The strategy improvement algorithm for reachability games of [4] is based on locally improving the strategy on the basis of the valuation it yields. This approach does not suffice for safety games: the sequence of strategies obtained would yield increasing values to player 1, but these value would not necessarily converge to the value of the game. In this paper, we introduce a novel, and non-local, improvement step, which augments the standard value-based improvement step. The non-local step involves the analysis of an appropriately-constructed turn-based game. As value iteration for safety games converges from above, while our sequences of strategies yields values that converge from below, the proof of convergence for our algorithm cannot be derived from a connection with value iteration, as was the case for reachability games. Thus, we developed new proof techniques to show both the monotonicity of the strategy values produced by our algorithm, and to show convergence to the value of the game.

## 2 Definitions

**Notation.** For a countable set  $A$ , a *probability distribution* on  $A$  is a function  $\delta : A \rightarrow [0, 1]$  such that  $\sum_{a \in A} \delta(a) = 1$ . We denote the set of probability distributions on  $A$  by  $\mathcal{D}(A)$ . Given a distribution  $\delta \in \mathcal{D}(A)$ , we denote by  $\text{Supp}(\delta) = \{x \in A \mid \delta(x) > 0\}$  the support set of  $\delta$ .

**Definition 1 (Concurrent games)** A (two-player) concurrent game structure  $G = \langle S, M, \Gamma_1, \Gamma_2, \delta \rangle$  consists of the following components:

- A finite state space  $S$  and a finite set  $M$  of moves or actions.
- Two move assignments  $\Gamma_1, \Gamma_2 : S \rightarrow 2^M \setminus \emptyset$ . For  $i \in \{1, 2\}$ , assignment  $\Gamma_i$  associates with each state  $s \in S$  a nonempty set  $\Gamma_i(s) \subseteq M$  of moves available to player  $i$  at state  $s$ .
- A probabilistic transition function  $\delta : S \times M \times M \rightarrow \mathcal{D}(S)$  that gives the probability  $\delta(s, a_1, a_2)(t)$  of a transition from  $s$  to  $t$  when player 1 chooses at state  $s$  move  $a_1$  and player 2 chooses move  $a_2$ , for all  $s, t \in S$  and  $a_1 \in \Gamma_1(s)$ ,  $a_2 \in \Gamma_2(s)$ .

We denote by  $|\delta|$  the size of transition function, i.e.,  $|\delta| = \sum_{s \in S, a \in \Gamma_1(s), b \in \Gamma_2(s), t \in S} |\delta(s, a, b)(t)|$ , where  $|\delta(s, a, b)(t)|$  is the number of bits required to specify the transition probability  $\delta(s, a, b)(t)$ . We denote by  $|G|$  the size of the game graph, and  $|G| = |\delta| + |S|$ . At every state  $s \in S$ , player 1 chooses a move  $a_1 \in \Gamma_1(s)$ , and simultaneously and independently player 2 chooses a move  $a_2 \in \Gamma_2(s)$ . The game then proceeds to the successor state  $t$  with probability  $\delta(s, a_1, a_2)(t)$ , for all  $t \in S$ . A state  $s$  is an *absorbing state* if for all  $a_1 \in \Gamma_1(s)$  and  $a_2 \in \Gamma_2(s)$ , we have  $\delta(s, a_1, a_2)(s) = 1$ . In other words, at an absorbing state  $s$  for all choices of moves of the two players, the successor state is always  $s$ .

**Definition 2 (Turn-based stochastic games)** A turn-based stochastic game graph ( $2^{1/2}$ -player game graph)  $G = \langle (S, E), (S_1, S_2, S_R), \delta \rangle$  consists of a finite directed graph  $(S, E)$ , a partition  $(S_1, S_2, S_R)$  of the finite set  $S$  of states, and a probabilistic transition function  $\delta: S_R \rightarrow \mathcal{D}(S)$ , where  $\mathcal{D}(S)$  denotes the set of probability distributions over the state space  $S$ . The states in  $S_1$  are the player-1 states, where player 1 decides the successor state; the states in  $S_2$  are the player-2 states, where player 2 decides the successor state; and the states in  $S_R$  are the random or probabilistic states, where the successor state is chosen according to the probabilistic transition function  $\delta$ . We assume that for  $s \in S_R$  and  $t \in S$ , we have  $(s, t) \in E$  iff  $\delta(s)(t) > 0$ , and we often write  $\delta(s, t)$  for  $\delta(s)(t)$ . For technical convenience we assume that every state in the graph  $(S, E)$  has at least one outgoing edge. For a state  $s \in S$ , we write  $E(s)$  to denote the set  $\{t \in S \mid (s, t) \in E\}$  of possible successors. We denote by  $|\delta|$  the size of the transition function, i.e.,  $|\delta| = \sum_{s \in S_R, t \in S} |\delta(s)(t)|$ , where  $|\delta(s)(t)|$  is the number of bits required to specify the transition probability  $\delta(s)(t)$ . We denote by  $|G|$  the size of the game graph, and  $|G| = |\delta| + |S| + |E|$ .

**Plays.** A play  $\omega$  of  $G$  is an infinite sequence  $\omega = \langle s_0, s_1, s_2, \dots \rangle$  of states in  $S$  such that for all  $k \geq 0$ , there are moves  $a_1^k \in \Gamma_1(s_k)$  and  $a_2^k \in \Gamma_2(s_k)$  with  $\delta(s_k, a_1^k, a_2^k)(s_{k+1}) > 0$ . We denote by  $\Omega$  the set of all plays, and by  $\Omega_s$  the set of all plays  $\omega = \langle s_0, s_1, s_2, \dots \rangle$  such that  $s_0 = s$ , that is, the set of plays starting from state  $s$ .

**Selectors and strategies.** A selector  $\xi$  for player  $i \in \{1, 2\}$  is a function  $\xi: S \rightarrow \mathcal{D}(M)$  such that for all states  $s \in S$  and moves  $a \in M$ , if  $\xi(s)(a) > 0$ , then  $a \in \Gamma_i(s)$ . A selector  $\xi$  for player  $i$  at a state  $s$  is a distribution over moves such that if  $\xi(s)(a) > 0$ , then  $a \in \Gamma_i(s)$ . We denote by  $\Lambda_i$  the set of all selectors for player  $i \in \{1, 2\}$ , and similarly, we denote by  $\Lambda_i(s)$  the set of all selectors for player  $i$  at a state  $s$ . The selector  $\xi$  is *pure* if for every state  $s \in S$ , there is a move  $a \in M$  such that  $\xi(s)(a) = 1$ . A *strategy* for player  $i \in \{1, 2\}$  is a function  $\pi: S^+ \rightarrow \mathcal{D}(M)$  that associates with every finite, nonempty sequence of states, representing the history of the play so far, a selector for player  $i$ ; that is, for all  $w \in S^*$  and  $s \in S$ , we have  $\text{Supp}(\pi(w \cdot s)) \subseteq \Gamma_i(s)$ . The strategy  $\pi$  is *pure* if it always chooses a pure selector; that is, for all  $w \in S^+$ , there is a move  $a \in M$  such that  $\pi(w)(a) = 1$ . A *memoryless* strategy is independent of the history of the play and depends only on the current state. Memoryless strategies correspond to selectors; we write  $\bar{\xi}$  for the memoryless strategy consisting in playing forever the selector  $\xi$ . A strategy is *pure memoryless* if it is both pure and memoryless. In a turn-based stochastic game, a strategy for player 1 is a function  $\pi_1: S^* \cdot S_1 \rightarrow \mathcal{D}(S)$ , such that for all  $w \in S^*$  and for all  $s \in S_1$  we have  $\text{Supp}(\pi_1(w \cdot s)) \subseteq E(s)$ . Memoryless strategies and pure memoryless strategies are obtained as the restriction of strategies as in the case of concurrent game graphs. The family of strategies for player 2 are defined analogously. We denote by  $\Pi_1$  and  $\Pi_2$  the sets of all strategies for player 1 and player 2, respectively. We denote by  $\Pi_i^M$  and  $\Pi_i^{PM}$  the sets of memoryless strategies and pure memoryless strategies for player  $i$ , respectively.

**Destinations of moves and selectors.** For all states  $s \in S$  and moves  $a_1 \in \Gamma_1(s)$  and  $a_2 \in \Gamma_2(s)$ , we indicate by  $\text{Dest}(s, a_1, a_2) = \text{Supp}(\delta(s, a_1, a_2))$  the set of possible successors of  $s$  when the moves  $a_1$  and  $a_2$  are chosen. Given a state  $s$ , and selectors  $\xi_1$  and  $\xi_2$  for the two players, we

denote by

$$Dest(s, \xi_1, \xi_2) = \bigcup_{\substack{a_1 \in Supp(\xi_1(s)), \\ a_2 \in Supp(\xi_2(s))}} Dest(s, a_1, a_2)$$

the set of possible successors of  $s$  with respect to the selectors  $\xi_1$  and  $\xi_2$ .

Once a starting state  $s$  and strategies  $\pi_1$  and  $\pi_2$  for the two players are fixed, the game is reduced to an ordinary stochastic process. Hence, the probabilities of events are uniquely defined, where an *event*  $\mathcal{A} \subseteq \Omega_s$  is a measurable set of plays. For an event  $\mathcal{A} \subseteq \Omega_s$ , we denote by  $\Pr_s^{\pi_1, \pi_2}(\mathcal{A})$  the probability that a play belongs to  $\mathcal{A}$  when the game starts from  $s$  and the players follow the strategies  $\pi_1$  and  $\pi_2$ . Similarly, for a measurable function  $f : \Omega_s \rightarrow \mathbb{R}$ , we denote by  $E_s^{\pi_1, \pi_2}(f)$  the expected value of  $f$  when the game starts from  $s$  and the players follow the strategies  $\pi_1$  and  $\pi_2$ . For  $i \geq 0$ , we denote by  $\Theta_i : \Omega \rightarrow S$  the random variable denoting the  $i$ -th state along a play.

**Valuations.** A *valuation* is a mapping  $v : S \rightarrow [0, 1]$  associating a real number  $v(s) \in [0, 1]$  with each state  $s$ . Given two valuations  $v, w : S \rightarrow \mathbb{R}$ , we write  $v \leq w$  when  $v(s) \leq w(s)$  for all states  $s \in S$ . For an event  $\mathcal{A}$ , we denote by  $\Pr^{\pi_1, \pi_2}(\mathcal{A})$  the valuation  $S \rightarrow [0, 1]$  defined for all states  $s \in S$  by  $(\Pr^{\pi_1, \pi_2}(\mathcal{A}))(s) = \Pr_s^{\pi_1, \pi_2}(\mathcal{A})$ . Similarly, for a measurable function  $f : \Omega_s \rightarrow [0, 1]$ , we denote by  $E^{\pi_1, \pi_2}(f)$  the valuation  $S \rightarrow [0, 1]$  defined for all  $s \in S$  by  $(E^{\pi_1, \pi_2}(f))(s) = E_s^{\pi_1, \pi_2}(f)$ .

**Reachability and safety objectives.** Given a set  $F \subseteq S$  of *safe* states, the objective of a safety game consists in never leaving  $F$ . Therefore, we define the set of winning plays as the set  $\text{Safe}(F) = \{\langle s_0, s_1, s_2, \dots \rangle \in \Omega \mid s_k \in F \text{ for all } k \geq 0\}$ . Given a subset  $T \subseteq S$  of *target* states, the objective of a reachability game consists in reaching  $T$ . Correspondingly, the set winning plays is  $\text{Reach}(T) = \{\langle s_0, s_1, s_2, \dots \rangle \in \Omega \mid s_k \in T \text{ for some } k \geq 0\}$  of plays that visit  $T$ . For all  $F \subseteq S$  and  $T \subseteq S$ , the sets  $\text{Safe}(F)$  and  $\text{Reach}(T)$  is measurable. An objective in general is a measurable set, and in this paper we would consider only reachability and safety objectives. For an objective  $\Phi$ , the probability of satisfying  $\Phi$  from a state  $s \in S$  under strategies  $\pi_1$  and  $\pi_2$  for players 1 and 2, respectively, is  $\Pr_s^{\pi_1, \pi_2}(\Phi)$ . We define the *value* for player 1 of game with objective  $\Phi$  from the state  $s \in S$  as

$$\langle\langle 1 \rangle\rangle_{\text{val}}(\Phi)(s) = \sup_{\pi_1 \in \Pi_1} \inf_{\pi_2 \in \Pi_2} \Pr_s^{\pi_1, \pi_2}(\Phi);$$

i.e., the value is the maximal probability with which player 1 can guarantee the satisfaction of  $\Phi$  against all player 2 strategies. Given a player-1 strategy  $\pi_1$ , we use the notation

$$\langle\langle 1 \rangle\rangle_{\text{val}}^{\pi_1}(\Phi)(s) = \inf_{\pi_2 \in \Pi_2} \Pr_s^{\pi_1, \pi_2}(\Phi).$$

A strategy  $\pi_1$  for player 1 is *optimal* for an objective  $\Phi$  if for all states  $s \in S$ , we have

$$\langle\langle 1 \rangle\rangle_{\text{val}}^{\pi_1}(\Phi)(s) = \langle\langle 1 \rangle\rangle_{\text{val}}(\Phi)(s).$$

For  $\varepsilon > 0$ , a strategy  $\pi_1$  for player 1 is  $\varepsilon$ -*optimal* if for all states  $s \in S$ , we have

$$\langle\langle 1 \rangle\rangle_{\text{val}}^{\pi_1}(\Phi)(s) \geq \langle\langle 1 \rangle\rangle_{\text{val}}(\Phi)(s) - \varepsilon.$$

The notion of values and optimal strategies for player 2 are defined analogously. Reachability and safety objectives are dual, i.e., we have  $\text{Reach}(T) = \Omega \setminus \text{Safe}(S \setminus T)$ . The quantitative determinacy result of [18] ensures that for all states  $s \in S$ , we have

$$\langle\langle 1 \rangle\rangle_{\text{val}}(\text{Safe}(F))(s) + \langle\langle 2 \rangle\rangle_{\text{val}}(\text{Reach}(S \setminus F))(s) = 1.$$

**Theorem 1 (Memoryless determinacy)** *For all concurrent game graphs  $G$ , for all  $F, T \subseteq S$ , such that  $F = S \setminus T$ , the following assertions hold.*

1. [14] *Memoryless optimal strategies exist for safety objectives  $\text{Safe}(F)$ .*
2. [4, 13] *For all  $\varepsilon > 0$ , memoryless  $\varepsilon$ -optimal strategies exist for reachability objectives  $\text{Reach}(T)$ .*
3. [5] *If  $G$  is a turn-based stochastic game graph, then pure memoryless optimal strategies exist for reachability objectives  $\text{Reach}(T)$  and safety objectives  $\text{Safe}(F)$ .*

### 3 Markov Decision Processes

To develop our arguments, we need some facts about one-player versions of concurrent stochastic games, known as *Markov decision processes* (MDPs) [12, 2]. For  $i \in \{1, 2\}$ , a *player- $i$  MDP* (for short,  $i$ -MDP) is a concurrent game where, for all states  $s \in S$ , we have  $|\Gamma_{3-i}(s)| = 1$ . Given a concurrent game  $G$ , if we fix a memoryless strategy corresponding to selector  $\xi_1$  for player 1, the game is equivalent to a 2-MDP  $G_{\xi_1}$  with the transition function

$$\delta_{\xi_1}(s, a_2)(t) = \sum_{a_1 \in \Gamma_1(s)} \delta(s, a_1, a_2)(t) \cdot \xi_1(s)(a_1),$$

for all  $s \in S$  and  $a_2 \in \Gamma_2(s)$ . Similarly, if we fix selectors  $\xi_1$  and  $\xi_2$  for both players in a concurrent game  $G$ , we obtain a Markov chain, which we denote by  $G_{\xi_1, \xi_2}$ .

**End components.** In an MDP, the sets of states that play an equivalent role to the closed recurrent classes of Markov chains [16] are called “end components” [7, 8].

**Definition 3 (End components)** *An end component of an  $i$ -MDP  $G$ , for  $i \in \{1, 2\}$ , is a subset  $C \subseteq S$  of the states such that there is a selector  $\xi$  for player  $i$  so that  $C$  is a closed recurrent class of the Markov chain  $G_\xi$ .*

It is not difficult to see that an equivalent characterization of an end component  $C$  is the following. For each state  $s \in C$ , there is a subset  $M_i(s) \subseteq \Gamma_i(s)$  of moves such that:

1. (*closed*) if a move in  $M_i(s)$  is chosen by player  $i$  at state  $s$ , then all successor states that are obtained with nonzero probability lie in  $C$ ; and
2. (*recurrent*) the graph  $(C, E)$ , where  $E$  consists of the transitions that occur with nonzero probability when moves in  $M_i(\cdot)$  are chosen by player  $i$ , is strongly connected.

Given a play  $\omega \in \Omega$ , we denote by  $\text{Inf}(\omega)$  the set of states that occurs infinitely often along  $\omega$ . Given a set  $\mathcal{F} \subseteq 2^S$  of subsets of states, we denote by  $\text{Inf}(\mathcal{F})$  the event  $\{\omega \mid \text{Inf}(\omega) \in \mathcal{F}\}$ . The following theorem states that in a 2-MDP, for every strategy of player 2, the set of states that are visited infinitely often is, with probability 1, an end component. Corollary 1 follows easily from Theorem 2.

**Theorem 2** [8] *For a player-1 selector  $\xi_1$ , let  $\mathcal{C}$  be the set of end components of a 2-MDP  $G_{\xi_1}$ . For all player-2 strategies  $\pi_2$  and all states  $s \in S$ , we have  $\Pr_s^{\xi_1, \pi_2}(\text{Inf}(\mathcal{C})) = 1$ .*

**Corollary 1** *For a player-1 selector  $\xi_1$ , let  $\mathcal{C}$  be the set of end components of a 2-MDP  $G_{\xi_1}$ , and let  $Z = \bigcup_{C \in \mathcal{C}} C$  be the set of states of all end components. For all player-2 strategies  $\pi_2$  and all states  $s \in S$ , we have  $\Pr_s^{\bar{\xi}_1, \pi_2}(\text{Reach}(Z)) = 1$ .*

**MDPs with reachability objectives.** Given a 2-MDP with a reachability objective  $\text{Reach}(T)$  for player 2, where  $T \subseteq S$ , the values can be obtained as the solution of a linear program [14]. The linear program has a variable  $x(s)$  for all states  $s \in S$ , and the objective function and the constraints are as follows:

$$\begin{aligned} \min \quad & \sum_{s \in S} x(s) \quad \text{subject to} \\ x(s) \geq \quad & \sum_{t \in S} x(t) \cdot \delta(s, a_2)(t) \quad \text{for all } s \in S \text{ and } a_2 \in \Gamma_2(s) \\ x(s) = 1 \quad & \text{for all } s \in T \\ 0 \leq x(s) \leq 1 \quad & \text{for all } s \in S \end{aligned}$$

The correctness of the above linear program to compute the values follows from [12, 14].

## 4 Strategy Improvement for Safety Games

In this section we present a strategy improvement algorithm for concurrent games with safety objectives. The algorithm will produce a sequence of selectors  $\gamma_0, \gamma_1, \gamma_2, \dots$  for player 1, such that:

1. for all  $i \geq 0$ , we have  $\langle\langle 1 \rangle\rangle_{\text{val}}^{\bar{\gamma}_i}(\text{Safe}(F)) \leq \langle\langle 1 \rangle\rangle_{\text{val}}^{\bar{\gamma}_{i+1}}(\text{Safe}(F))$ ;
2. if there is  $i \geq 0$  such that  $\gamma_i = \gamma_{i+1}$ , then  $\langle\langle 1 \rangle\rangle_{\text{val}}^{\bar{\gamma}_i}(\text{Safe}(F)) = \langle\langle 1 \rangle\rangle_{\text{val}}(\text{Safe}(F))$ ; and
3.  $\lim_{i \rightarrow \infty} \langle\langle 1 \rangle\rangle_{\text{val}}^{\bar{\gamma}_i}(\text{Safe}(F)) = \langle\langle 1 \rangle\rangle_{\text{val}}(\text{Safe}(F))$ .

Condition 1 guarantees that the algorithm computes a sequence of monotonically improving selectors. Condition 2 guarantees that if a selector cannot be improved, then it is optimal. Condition 3 guarantees that the value guaranteed by the selectors converges to the value of the game, or equivalently, that for all  $\varepsilon > 0$ , there is a number  $i$  of iterations such that the memoryless player-1 strategy  $\bar{\gamma}_i$  is  $\varepsilon$ -optimal. Note that for concurrent safety games, there may be no  $i \geq 0$  such that  $\gamma_i = \gamma_{i+1}$ , that is, the algorithm may fail to generate an optimal selector. This is because there are concurrent safety games such that the values are irrational [11]. We start with a few notations

**The  $Pre$  operator and optimal selectors.** Given a valuation  $v$ , and two selectors  $\xi_1 \in \Lambda_1$  and  $\xi_2 \in \Lambda_2$ , we define the valuations  $Pre_{\xi_1, \xi_2}(v)$ ,  $Pre_{1, \xi_1}(v)$ , and  $Pre_1(v)$  as follows, for all states  $s \in S$ :

$$\begin{aligned} Pre_{\xi_1, \xi_2}(v)(s) &= \sum_{a, b \in M} \sum_{t \in S} v(t) \cdot \delta(s, a, b)(t) \cdot \xi_1(s)(a) \cdot \xi_2(s)(b) \\ Pre_{1, \xi_1}(v)(s) &= \inf_{\xi_2 \in \Lambda_2} Pre_{\xi_1, \xi_2}(v)(s) \\ Pre_1(v)(s) &= \sup_{\xi_1 \in \Lambda_1} \inf_{\xi_2 \in \Lambda_2} Pre_{\xi_1, \xi_2}(v)(s) \end{aligned}$$

Intuitively,  $Pre_1(v)(s)$  is the greatest expectation of  $v$  that player 1 can guarantee at a successor state of  $s$ . Also note that given a valuation  $v$ , the computation of  $Pre_1(v)$  reduces to the solution of a zero-sum one-shot matrix game, and can be solved by linear programming. Similarly,  $Pre_{1:\xi_1}(v)(s)$  is the greatest expectation of  $v$  that player 1 can guarantee at a successor state of  $s$  by playing the selector  $\xi_1$ . Note that all of these operators on valuations are monotonic: for two valuations  $v, w$ , if  $v \leq w$ , then for all selectors  $\xi_1 \in \Lambda_1$  and  $\xi_2 \in \Lambda_2$ , we have  $Pre_{\xi_1, \xi_2}(v) \leq Pre_{\xi_1, \xi_2}(w)$ ,  $Pre_{1:\xi_1}(v) \leq Pre_{1:\xi_1}(w)$ , and  $Pre_1(v) \leq Pre_1(w)$ . Given a valuation  $v$  and a state  $s$ , we define by

$$\text{OptSel}(v, s) = \{\xi_1 \in \Lambda_1(s) \mid Pre_{1:\xi_1}(v)(s) = Pre_1(v)(s)\}$$

the set of optimal selectors for  $v$  at state  $s$ . For an optimal selector  $\xi_1 \in \text{OptSel}(v, s)$ , we define the set of counter-optimal actions as follows:

$$\text{CountOpt}(v, s, \xi_1) = \{b \in \Gamma_2(s) \mid Pre_{\xi_1, b}(v)(s) = Pre_1(v)(s)\}.$$

Observe that for  $\xi_1 \in \text{OptSel}(v, s)$ , for all  $b \in \Gamma_2(s) \setminus \text{CountOpt}(v, s, \xi_1)$  we have  $Pre_{\xi_1, b}(v)(s) > Pre_1(v)(s)$ . We define the set of optimal selector support and the counter-optimal action set as follows:

$$\begin{aligned} \text{OptSelCount}(v, s) &= \{(A, B) \subseteq \Gamma_1(s) \times \Gamma_2(s) \mid \exists \xi_1 \in \Lambda_1(s). \xi_1 \in \text{OptSel}(v, s) \\ &\quad \wedge \text{Supp}(\xi_1) = A \wedge \text{CountOpt}(v, s, \xi_1) = B\}; \end{aligned}$$

i.e., it consists of pairs  $(A, B)$  of actions of player 1 and player 2, such that there is an optimal selector  $\xi_1$  with support  $A$ , and  $B$  is the set of counter-optimal actions to  $\xi_1$ .

**Turn-based reduction.** Given a concurrent game  $G = \langle S, M, \Gamma_1, \Gamma_2, \delta \rangle$  and a valuation  $v$  we construct a turn-based stochastic game  $\overline{G}_v = \langle (\overline{S}, \overline{E}), (\overline{S}_1, \overline{S}_2, \overline{S}_R), \overline{\delta} \rangle$  as follows:

1. The set of states is as follows:

$$\begin{aligned} \overline{S} &= S \cup \{(s, A, B) \mid s \in S, (A, B) \in \text{OptSelCount}(v, s)\} \\ &\quad \cup \{(s, A, b) \mid s \in S, (A, B) \in \text{OptSelCount}(v, s), b \in B\}. \end{aligned}$$

2. The state space partition is as follows:  $\overline{S}_1 = S$ ;  $\overline{S}_2 = \{(s, A, B) \mid s \in S, (A, B) \in \text{OptSelCount}(v, s)\}$ ; and  $\overline{S}_R = \overline{S} \setminus (\overline{S}_1 \cup \overline{S}_2)$ .

3. The set of edges is as follows:

$$\begin{aligned} \overline{E} &= \{(s, (s, A, B)) \mid s \in S, (A, B) \in \text{OptSelCount}(v, s)\} \\ &\quad \cup \{((s, A, B), (s, A, b)) \mid b \in B\} \cup \{((s, A, b), t) \mid t \in \bigcup_{a \in A} \text{Dest}(s, a, b)\}. \end{aligned}$$

4. The transition function  $\overline{\delta}$  for all states in  $\overline{S}_R$  is uniform over its successors.

Intuitively, the reduction is as follows. Given the valuation  $v$ , state  $s$  is a player 1 state where player 1 can select a pair  $(A, B)$  (and move to state  $(s, A, B)$ ) with  $A \subseteq \Gamma_1(s)$  and  $B \subseteq \Gamma_2(s)$  such that there is an optimal selector  $\xi_1$  with support exactly  $A$  and the set of counter-optimal actions to  $\xi_1$  is the set  $B$ . From a player 2 state  $(s, A, B)$ , player 2 can choose any action  $b$  from the set  $B$ , and move to state  $(s, A, b)$ . A state  $(s, A, b)$  is a probabilistic state where all the states in  $\bigcup_{a \in A} \text{Dest}(s, a, b)$  are chosen uniformly at random. Given a set  $F \subseteq S$  we denote by  $\overline{F} = F \cup \{(s, A, B) \in \overline{S} \mid s \in F\} \cup \{(s, A, b) \in \overline{S} \mid s \in F\}$ . We refer to the above reduction as TB, i.e.,  $(\overline{G}_v, \overline{F}) = \text{TB}(G, v, F)$ .

**Value-class of a valuation.** Given a valuation  $v$  and a real  $0 \leq r \leq 1$ , the *value-class*  $U_r(v)$  of value  $r$  is the set of states with valuation  $r$ , i.e.,  $U_r(v) = \{s \in S \mid v(s) = r\}$



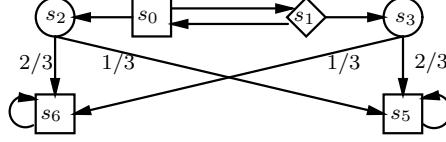


Figure 1: A turn-based stochastic safety game.

#### 4.1 The strategy improvement algorithm

**Ordering of strategies.** Let  $G$  be a concurrent game and  $F$  be the set of safe states. Let  $T = S \setminus F$ . Given a concurrent game graph  $G$  with a safety objective  $\text{Safe}(F)$ , the set of *almost-sure winning* states is the set of states  $s$  such that the value at  $s$  is 1, i.e.,  $W_1 = \{s \in S \mid \langle\langle 1 \rangle\rangle_{\text{val}}(\text{Safe}(F)) = 1\}$  is the set of almost-sure winning states. An optimal strategy from  $W_1$  is referred as an almost-sure winning strategy. The set  $W_1$  and an almost-sure winning strategy can be computed in linear time by the algorithm given in [9]. We assume without loss of generality that all states in  $W_1 \cup T$  are absorbing. We define a preorder  $\prec$  on the strategies for player 1 as follows: given two player 1 strategies  $\pi_1$  and  $\pi'_1$ , let  $\pi_1 \prec \pi'_1$  if the following two conditions hold: (i)  $\langle\langle 1 \rangle\rangle_{\text{val}}^{\pi_1}(\text{Safe}(F)) \leq \langle\langle 1 \rangle\rangle_{\text{val}}^{\pi'_1}(\text{Safe}(F))$ ; and (ii)  $\langle\langle 1 \rangle\rangle_{\text{val}}^{\pi_1}(\text{Safe}(F))(s) < \langle\langle 1 \rangle\rangle_{\text{val}}^{\pi'_1}(\text{Safe}(F))(s)$  for some state  $s \in S$ . Furthermore, we write  $\pi_1 \preceq \pi'_1$  if either  $\pi_1 \prec \pi'_1$  or  $\pi_1 = \pi'_1$ . We first present an example that shows the improvements based only on  $\text{Pre}_1$  operators are not sufficient for safety games, even on turn-based games and then present our algorithm.

**Example 1** Consider the turn-based stochastic game shown in Fig 1, where the  $\square$  states are player 1 states, the  $\diamond$  states are player 2 states, and  $\circ$  states are random states with probabilities labeled on edges. The safety goal is to avoid the state  $s_6$ . Consider a memoryless strategy  $\pi_1$  for player 1 that chooses the successor  $s_0 \rightarrow s_2$ , and the counter-strategy  $\pi_2$  for player 2 chooses  $s_1 \rightarrow s_0$ . Given the strategies  $\pi_1$  and  $\pi_2$ , the value at  $s_0, s_1$  and  $s_2$  is  $1/3$ , and since all successors of  $s_0$  have value  $1/3$ , the value cannot be improved by  $\text{Pre}_1$ . However, note that if player 2 is restricted to choose only value optimal selectors for the value  $1/3$ , then player 1 can switch to the strategy  $s_0 \rightarrow s_2$  and ensure that the game stays in the value class  $1/3$  with probability 1. Hence switching to  $s_0 \rightarrow s_2$  would force player 2 to select a counter-strategy that switches to the strategy  $s_1 \rightarrow s_3$ , and thus player 1 can get a value  $2/3$ . ■

**Informal description of Algorithm 1.** We now present the strategy improvement algorithm (Algorithm 1) for computing the values for all states in  $S \setminus W_1$ . The algorithm iteratively improves player-1 strategies according to the preorder  $\prec$ . The algorithm starts with the random selector  $\gamma_0 = \bar{\xi}_1^{\text{unif}}$  that plays at all states all actions uniformly at random. At iteration  $i+1$ , the algorithm considers the memoryless player-1 strategy  $\bar{\gamma}_i$  and computes the value  $\langle\langle 1 \rangle\rangle_{\text{val}}^{\bar{\gamma}_i}(\text{Safe}(F))$ . Observe that since  $\bar{\gamma}_i$  is a memoryless strategy, the computation of  $\langle\langle 1 \rangle\rangle_{\text{val}}^{\bar{\gamma}_i}(\text{Safe}(F))$  involves solving the 2-MDP  $G_{\bar{\gamma}_i}$ . The valuation  $\langle\langle 1 \rangle\rangle_{\text{val}}^{\bar{\gamma}_i}(\text{Safe}(F))$  is named  $v_i$ . For all states  $s$  such that  $\text{Pre}_1(v_i)(s) > v_i(s)$ , the memoryless strategy at  $s$  is modified to a selector that is value-optimal for  $v_i$ . The algorithm then proceeds to the next iteration. If  $\text{Pre}_1(v_i) = v_i$ , then the algorithm constructs the game  $(\bar{G}_{v_i}, \bar{F}) = \text{TB}(G, v_i, F)$ , and computes  $\bar{A}_i$  as the set of almost-sure winning states in  $\bar{G}_{v_i}$  for the objective  $\text{Safe}(\bar{F})$ . Let  $U = (\bar{A}_i \cap S) \setminus W_1$ . If  $U$  is non-empty, then a selector  $\gamma_{i+1}$  is obtained at  $U$

---

**Algorithm 1** Safety Strategy-Improvement Algorithm
 

---

**Input:** a concurrent game structure  $G$  with safe set  $F$ .  
**Output:** a strategy  $\bar{\gamma}$  for player 1.

0. Compute  $W_1 = \{s \in S \mid \langle\langle 1 \rangle\rangle_{\text{val}}(\text{Safe}(F))(s) = 1\}$ .
1. Let  $\gamma_0 = \xi_1^{\text{unif}}$  and  $i = 0$ .
2. Compute  $v_0 = \langle\langle 1 \rangle\rangle_{\text{val}}^{\bar{\gamma}_0}(\text{Safe}(F))$ .
3. **do** {
  - 3.1. Let  $I = \{s \in S \setminus (W_1 \cup T) \mid \text{Pre}_1(v_i)(s) > v_i(s)\}$ .
  - 3.2 **if**  $I \neq \emptyset$ , **then**
    - 3.2.1 Let  $\xi_1$  be a player-1 selector such that for all states  $s \in I$ , we have  $\text{Pre}_{1;\xi_1}(v_i)(s) = \text{Pre}_1(v_i)(s) > v_i(s)$ .
    - 3.2.2 The player-1 selector  $\gamma_{i+1}$  is defined as follows: for each state  $t \in S$ , let
 
$$\gamma_{i+1}(t) = \begin{cases} \gamma_i(t) & \text{if } s \notin I; \\ \xi_1(s) & \text{if } s \in I. \end{cases}$$
  - 3.3 **else**
    - 3.3.1 let  $(\bar{G}_{v_i}, \bar{F}) = \text{TB}(G, v_i, F)$
    - 3.3.2 let  $\bar{A}_i$  be the set of almost-sure winning states in  $\bar{G}_{v_i}$  for  $\text{Safe}(\bar{F})$  and  $\bar{\pi}_1$  be a pure memoryless almost-sure winning strategy from the set  $\bar{A}_i$ .
    - 3.3.3 **if**  $(\bar{A}_i \cap S) \setminus W_1 \neq \emptyset$ 
      - 3.3.3.1 let  $U = (\bar{A}_i \cap S) \setminus W_1$
      - 3.3.3.2 The player-1 selector  $\gamma_{i+1}$  is defined as follows: for  $t \in S$ , let
 
$$\gamma_{i+1}(t) = \begin{cases} \gamma_i(t) & \text{if } s \notin U; \\ \xi_1(s) & \text{if } s \in U, \xi_1(s) \in \text{OptSel}(v_i, s), \\ \bar{\pi}_1(s) & \text{if } s \in U, \xi_1(s) \notin \text{OptSel}(v_i, s). \end{cases}$$
  - 3.4. Compute  $v_{i+1} = \langle\langle 1 \rangle\rangle_{\text{val}}^{\bar{\gamma}_{i+1}}(\text{Safe}(F))$ .
  - 3.5. Let  $i = i + 1$ .
- until**  $I = \emptyset$  and  $(\bar{A}_{i-1} \cap S) \setminus W_1 = \emptyset$ .
4. **return**  $\bar{\gamma}_i$ .

---

from an pure memoryless optimal strategy (i.e., an almost-sure winning strategy) in  $\bar{G}_{v_i}$ , and the algorithm proceeds to iteration  $i + 1$ . If  $\text{Pre}_1(v_i) = v_i$  and  $U$  is empty, then the algorithm stops and returns the memoryless strategy  $\bar{\gamma}_i$  for player 1. Unlike strategy improvement algorithms for turn-based games (see [6] for a survey), Algorithm 1 is not guaranteed to terminate, because the value of a safety game may not be rational.

**Lemma 1** *Let  $\gamma_i$  and  $\gamma_{i+1}$  be the player-1 selectors obtained at iterations  $i$  and  $i + 1$  of Algorithm 1. Let  $I = \{s \in S \setminus (W_1 \cup T) \mid \text{Pre}_1(v_i)(s) > v_i(s)\}$ . Let  $v_i = \langle\langle 1 \rangle\rangle_{\text{val}}^{\bar{\gamma}_i}(\text{Safe}(F))$  and  $v_{i+1} = \langle\langle 1 \rangle\rangle_{\text{val}}^{\bar{\gamma}_{i+1}}(\text{Safe}(F))$ . Then  $v_{i+1}(s) \geq \text{Pre}_1(v_i)(s)$  for all states  $s \in S$ ; and therefore  $v_{i+1}(s) \geq v_i(s)$  for all states  $s \in S$ , and  $v_{i+1}(s) > v_i(s)$  for all states  $s \in I$ .*

**Proof.** Consider the valuations  $v_i$  and  $v_{i+1}$  obtained at iterations  $i$  and  $i + 1$ , respectively, and let  $w_i$  be the valuation defined by  $w_i(s) = 1 - v_i(s)$  for all states  $s \in S$ . The counter-optimal strategy

for player 2 to minimize  $v_{i+1}$  is obtained by maximizing the probability to reach  $T$ . Let

$$w_{i+1}(s) = \begin{cases} w_i(s) & \text{if } s \in S \setminus I; \\ 1 - \text{Pre}_1(v_i)(s) < w_i(s) & \text{if } s \in I. \end{cases}$$

In other words,  $w_{i+1} = 1 - \text{Pre}_1(v_i)$ , and we also have  $w_{i+1} \leq w_i$ . We now show that  $w_{i+1}$  is a feasible solution to the linear program for MDPs with the objective  $\text{Reach}(T)$ , as described in Section 3. Since  $v_i = \langle\langle 1 \rangle\rangle_{\text{val}}^{\bar{\gamma}_i}(\text{Safe}(F))$ , it follows that for all states  $s \in S$  and all moves  $a_2 \in \Gamma_2(s)$ , we have

$$w_i(s) \geq \sum_{t \in S} w_i(t) \cdot \delta_{\gamma_i}(s, a_2).$$

For all states  $s \in S \setminus I$ , we have  $\gamma_i(s) = \gamma_{i+1}(s)$  and  $w_{i+1}(s) = w_i(s)$ , and since  $w_{i+1} \leq w_i$ , it follows that for all states  $s \in S \setminus I$  and all moves  $a_2 \in \Gamma_2(s)$ , we have

$$w_{i+1}(s) = w_i(s) \geq \sum_{t \in S} w_{i+1}(t) \cdot \delta_{\gamma_{i+1}}(s, a_2) \quad (\text{ for } s \in S \setminus I).$$

Since for  $s \in I$  the selector  $\gamma_{i+1}(s)$  is obtained as an optimal selector for  $\text{Pre}_1(v_i)(s)$ , it follows that for all states  $s \in I$  and all moves  $a_2 \in \Gamma_2(s)$ , we have

$$\text{Pre}_{\gamma_{i+1}, a_2}(v_i)(s) \geq \text{Pre}_1(v_i)(s);$$

in other words,  $1 - \text{Pre}_1(v_i)(s) \geq 1 - \text{Pre}_{\gamma_{i+1}, a_2}(v_i)(s)$ . Hence for all states  $s \in I$  and all moves  $a_2 \in \Gamma_2(s)$ , we have

$$w_{i+1}(s) \geq \sum_{t \in S} w_i(t) \cdot \delta_{\gamma_{i+1}}(s, a_2).$$

Since  $w_{i+1} \leq w_i$ , for all states  $s \in I$  and all moves  $a_2 \in \Gamma_2(s)$ , we have

$$w_{i+1}(s) \geq \sum_{t \in S} w_{i+1}(t) \cdot \delta_{\gamma_{i+1}}(s, a_2) \quad (\text{ for } s \in I).$$

Hence it follows that  $w_{i+1}$  is a feasible solution to the linear program for MDPs with reachability objectives. Since the reachability valuation for player 2 for  $\text{Reach}(T)$  is the least solution (observe that the objective function of the linear program is a minimizing function), it follows that  $v_{i+1} \geq 1 - w_{i+1} = \text{Pre}_1(v_i)$ . Thus we obtain  $v_{i+1}(s) \geq v_i(s)$  for all states  $s \in S$ , and  $v_{i+1}(s) > v_i(s)$  for all states  $s \in I$ . ■

Recall that by Example 1 it follows that improvement by only step 3.2 is not sufficient to guarantee convergence to optimal values. We now present a lemma about the turn-based reduction, and then show that step 3.3 also leads to an improvement. Finally, in Theorem 4 we show that if improvements by step 3.2 and step 3.3 are not possible, then the optimal value and an optimal strategy is obtained.

**Lemma 2** *Let  $G$  be a concurrent game with a set  $F$  of safe states. Let  $v$  be a valuation and consider  $(\bar{G}_v, \bar{F}) = \text{TB}(G, v, F)$ . Let  $\bar{A}$  be the set of almost-sure winning states in  $\bar{G}_v$  for the objective  $\text{Safe}(\bar{F})$ , and let  $\bar{\pi}_1$  be a pure memoryless almost-sure winning strategy from  $\bar{A}$  in  $\bar{G}_v$ . Consider a memoryless strategy  $\pi_1$  in  $G$  for states in  $\bar{A} \cap S$  as follows: if  $\bar{\pi}_1(s) = (s, A, B)$ , then  $\pi_1(s) \in \text{OptSel}(v, s)$  such that  $\text{Supp}(\pi_1(s)) = A$  and  $\text{OptSelCount}(v, s, \pi_1(s)) = B$ . Consider a pure memoryless strategy  $\pi_2$  for player 2. If for all states  $s \in \bar{A} \cap S$ , we have  $\pi_2(s) \in \text{OptSelCount}(v, s, \pi_1(s))$ , then for all  $s \in \bar{A} \cap S$ , we have  $\text{Pr}_s^{\pi_1, \pi_2}(\text{Safe}(F)) = 1$ .*

**Proof.** We analyze the Markov chain arising after the player fixes the memoryless strategies  $\pi_1$  and  $\pi_2$ . Given the strategy  $\pi_2$  consider the strategy  $\bar{\pi}_2$  as follows: if  $\bar{\pi}_1(s) = (s, A, B)$  and  $\pi_2(s) = b \in \text{OptSelCount}(v, s, \pi_1(s))$ , then at state  $(s, A, B)$  choose the successor  $(s, A, b)$ . Since  $\bar{\pi}_1$  is an almost-sure winning strategy for  $\text{Safe}(\bar{F})$ , it follows that in the Markov chain obtained by fixing  $\bar{\pi}_1$  and  $\bar{\pi}_2$  in  $\bar{G}_v$ , all closed connected recurrent set of states that intersect with  $\bar{A}$  are contained in  $\bar{A}$ , and from all states of  $\bar{A}$  the closed connected recurrent set of states within  $\bar{A}$  are reached with probability 1. It follows that in the Markov chain obtained from fixing  $\pi_1$  and  $\pi_2$  in  $G$  all closed connected recurrent set of states that intersect with  $\bar{A} \cap S$  are contained in  $\bar{A} \cap S$ , and from all states of  $\bar{A} \cap S$  the closed connected recurrent set of states within  $\bar{A} \cap S$  are reached with probability 1. The desired result follows. ■

**Lemma 3** *Let  $\gamma_i$  and  $\gamma_{i+1}$  be the player-1 selectors obtained at iterations  $i$  and  $i+1$  of Algorithm 1. Let  $I = \{s \in S \setminus (W_1 \cup T) \mid \text{Pre}_1(v_i)(s) > v_i(s)\} = \emptyset$ , and  $(\bar{A}_i \cap S) \setminus W_1 \neq \emptyset$ . Let  $v_i = \langle\langle 1 \rangle\rangle_{\text{val}}^{\bar{\gamma}_i}(\text{Safe}(F))$  and  $v_{i+1} = \langle\langle 1 \rangle\rangle_{\text{val}}^{\bar{\gamma}_{i+1}}(\text{Safe}(F))$ . Then  $v_{i+1}(s) \geq v_i(s)$  for all states  $s \in S$ , and  $v_{i+1}(s) > v_i(s)$  for some state  $s \in (\bar{A}_i \cap S) \setminus W_1$ .*

**Proof.** We first show that  $v_{i+1} \geq v_i$ . Let  $U = (\bar{A}_i \cap S) \setminus W_1$ . Let  $w_i(s) = 1 - v_i(s)$  for all states  $s \in S$ . Since  $v_i = \langle\langle 1 \rangle\rangle_{\text{val}}^{\bar{\gamma}_i}(\text{Safe}(F))$ , it follows that for all states  $s \in S$  and all moves  $a_2 \in \Gamma_2(s)$ , we have

$$w_i(s) \geq \sum_{t \in S} w_i(t) \cdot \delta_{\gamma_i}(s, a_2).$$

The selector  $\xi_1(s)$  chosen for  $\gamma_{i+1}$  at  $s \in U$  satisfies that  $\xi_1(s) \in \text{OptSel}(v_i, s)$ . It follows that for all states  $s \in S$  and all moves  $a_2 \in \Gamma_2(s)$ , we have

$$w_i(s) \geq \sum_{t \in S} w_i(t) \cdot \delta_{\gamma_{i+1}}(s, a_2).$$

It follows that the maximal probability with which player 2 can reach  $T$  against the strategy  $\bar{\gamma}_{i+1}$  is at most  $w_i$ . It follows that  $v_i(s) \leq v_{i+1}(s)$ .

We now argue that for some state  $s \in U$  we have  $v_{i+1}(s) > v_i(s)$ . Given the strategy  $\bar{\gamma}_{i+1}$ , consider a pure memoryless counter-optimal strategy  $\pi_2$  for player 2 to reach  $T$ . Since the selectors  $\gamma_{i+1}(s)$  at states  $s \in U$  are obtained from the almost-sure strategy  $\bar{\pi}$  in the turn-based game  $\bar{G}_{v_i}$  to satisfy  $\text{Safe}(\bar{F})$ , it follows from Lemma 2 that if for every state  $s \in U$ , the action  $\pi_2(s) \in \text{OptSelCount}(v_i, s, \gamma_{i+1})$ , then from all states  $s \in U$ , the game stays safe in  $F$  with probability 1. Since  $\bar{\gamma}_{i+1}$  is a given strategy for player 1, and  $\pi_2$  is counter-optimal against  $\bar{\gamma}_{i+1}$ , this would imply that  $U \subseteq \{s \in S \mid \langle\langle 1 \rangle\rangle_{\text{val}}(\text{Safe}(F)) = 1\}$ . This would contradict that  $W_1 = \{s \in S \mid \langle\langle 1 \rangle\rangle_{\text{val}}(\text{Safe}(F)) = 1\}$  and  $U \cap W_1 = \emptyset$ . It follows that for some state  $s^* \in U$  we have  $\pi_2(s^*) \notin \text{OptSelCount}(v_i, s^*, \gamma_{i+1})$ , and since  $\gamma_{i+1}(s^*) \in \text{OptSel}(v_i, s^*)$  we have

$$v_i(s^*) < \sum_{t \in S} v_i(t) \cdot \delta_{\gamma_{i+1}}(s^*, \pi_2(s^*));$$

in other words, we have

$$w_i(s^*) > \sum_{t \in S} w_i(t) \cdot \delta_{\gamma_{i+1}}(s^*, \pi_2(s^*)).$$

Define a valuation  $z$  as follows:  $z(s) = w_i(s)$  for  $s \neq s^*$ , and  $z(s^*) = \sum_{t \in S} w_i(t) \cdot \delta_{\gamma_{i+1}}(s^*, \pi_2(s^*))$ . Hence  $z < w_i$ , and given the strategy  $\bar{\gamma}_{i+1}$  and the counter-optimal strategy  $\pi_2$ , the valuation  $z$  satisfies the inequalities of the linear-program for reachability to  $T$ . It follows that the probability to reach  $T$  given  $\bar{\gamma}_{i+1}$  is at most  $z$ . Since  $z < w_i$ , it follows that  $v_{i+1}(s) \geq v_i(s)$  for all  $s \in S$ , and  $v_{i+1}(s^*) > v_i(s^*)$ . This concludes the proof. ■

We obtain the following theorem from Lemma 1 and Lemma 3 that shows that the sequences of values we obtain is monotonically non-decreasing.

**Theorem 3 (Monotonicity of values)** *For  $i \geq 0$ , let  $\gamma_i$  and  $\gamma_{i+1}$  be the player-1 selectors obtained at iterations  $i$  and  $i + 1$  of Algorithm 1. If  $\gamma_i \neq \gamma_{i+1}$ , then  $\langle\langle 1 \rangle\rangle_{\text{val}}^{\bar{\gamma}_i}(\text{Safe}(F)) < \langle\langle 1 \rangle\rangle_{\text{val}}^{\bar{\gamma}_{i+1}}(\text{Safe}(F))$ .*

**Theorem 4 (Optimality on termination)** *Let  $v_i$  be the valuation at iteration  $i$  of Algorithm 1 such that  $v_i = \langle\langle 1 \rangle\rangle_{\text{val}}^{\bar{\gamma}_i}(\text{Safe}(F))$ . If  $I = \{s \in S \setminus (W_1 \cup T) \mid \text{Pre}_1(v_i)(s) > v_i(s)\} = \emptyset$ , and  $(\bar{A}_i \cap S) \setminus W_1 = \emptyset$ , then  $\bar{\gamma}_i$  is an optimal strategy and  $v_i = \langle\langle 1 \rangle\rangle_{\text{val}}(\text{Safe}(F))$ .*

**Proof.** We show that for all memoryless strategies  $\pi_1$  for player 1 we have  $\langle\langle 1 \rangle\rangle_{\text{val}}^{\pi_1}(\text{Safe}(F)) \leq v_i$ . Since memoryless optimal strategies exist for concurrent games with safety objectives (Theorem 1) the desired result follows.

Let  $\bar{\pi}_2$  be a pure memoryless optimal strategy for player 2 in  $\bar{G}_{v_i}$  for the objective complementary to  $\text{Safe}(\bar{F})$ , where  $(\bar{G}_{v_i}, \text{Safe}(\bar{F})) = \text{TB}(G, v_i, F)$ . Consider a memoryless strategy  $\pi_1$  for player 1, and we define a pure memoryless strategy  $\pi_2$  for player 2 as follows.

1. If  $\pi_1(s) \notin \text{OptSel}(v_i, s)$ , then  $\pi_2(s) = b \in \Gamma_2(s)$ , such that  $\text{Pre}_{\pi_1(s), b}(v_i)(s) < v_i(s)$ ; (such a  $b$  exists since  $\pi_1(s) \notin \text{OptSel}(v_i, s)$ ).
2. If  $\pi_1(s) \in \text{OptSel}(v_i, s)$ , then let  $A = \text{Supp}(\pi_1(s))$ , and consider  $B$  such that  $B = \text{OptSelCount}(v_i, s, \pi_1(s))$ . Then we have  $\pi_2(s) = b$ , such that  $\bar{\pi}_2((s, A, B)) = (s, A, b)$ .

Observe that by construction of  $\pi_2$ , for all  $s \in S \setminus (W_1 \cup T)$ , we have  $\text{Pre}_{\pi_1(s), \pi_2(s)}(v_i)(s) \leq v_i(s)$ . We first show that in the Markov chain obtained by fixing  $\pi_1$  and  $\pi_2$  in  $G$ , there is no closed connected recurrent set of states  $C$  such that  $C \subseteq S \setminus (W_1 \cup T)$ . Assume towards contradiction that  $C$  is a closed connected recurrent set of states in  $S \setminus (W_1 \cup T)$ . The following case analysis achieves the contradiction.

1. Suppose for every state  $s \in C$  we have  $\pi_1(s) \in \text{OptSel}(v_i, s)$ . Then consider the strategy  $\bar{\pi}_1$  in  $\bar{G}_{v_i}$  such that for a state  $s \in C$  we have  $\bar{\pi}_1(s) = (s, A, B)$ , where  $\pi_1(s) = A$ , and  $B = \text{OptSelCount}(v_i, s, \pi_1(s))$ . Since  $C$  is closed connected recurrent states, it follows by construction that for all states  $s \in C$  in the game  $\bar{G}_{v_i}$  we have  $\text{Pr}_s^{\bar{\pi}_1, \bar{\pi}_2}(\text{Safe}(\bar{C})) = 1$ , where  $\bar{C} = C \cup \{(s, A, B) \mid s \in C\} \cup \{(s, A, b) \mid s \in C\}$ . It follows that for all  $s \in C$  in  $\bar{G}_{v_i}$  we have  $\text{Pr}_s^{\bar{\pi}_1, \bar{\pi}_2}(\text{Safe}(\bar{F})) = 1$ . Since  $\bar{\pi}_2$  is an optimal strategy, it follows that  $C \subseteq (\bar{A}_i \cap S) \setminus W_1$ . This contradicts that  $(\bar{A}_i \cap S) \setminus W_1 = \emptyset$ .
2. Otherwise for some state  $s^* \in C$  we have  $\pi_1(s^*) \notin \text{OptSel}(v_i, s^*)$ . Let  $r = \min\{q \mid U_q(v_i) \cap C \neq \emptyset\}$ , i.e.,  $r$  is the least value-class with non-empty intersection with  $C$ . Hence it follows that for all  $q < r$ , we have  $U_q(v_i) \cap C = \emptyset$ . Observe that since for all  $s \in C$  we have  $\text{Pre}_{\pi_1(s), \pi_2(s)}(v_i)(s) \leq v_i(s)$ , it follows that for all  $s \in U_r(v_i)$  either (a)  $\text{Dest}(s, \pi_1(s), \pi_2(s)) \subseteq$

$U_r(v_i)$ ; or (b)  $Dest(s, \pi_1(s), \pi_2(s)) \cap U_q(v_i) \neq \emptyset$ , for some  $q < r$ . Since  $U_r(v_i)$  is the least value-class with non-empty intersection with  $C$ , it follows that for all  $s \in U_r(v_i)$  we have  $Dest(s, \pi_1(s), \pi_2(s)) \subseteq U_r(v_i)$ . It follows that  $C \subseteq U_r(v_i)$ . Consider the state  $s^* \in C$  such that  $\pi_1(s^*) \notin \text{OptSel}(v_i, s)$ . By the construction of  $\pi_2(s)$ , we have  $Pre_{\pi_1(s^*), \pi_2(s^*)}(v_i)(s^*) < v_i(s^*)$ . Hence we must have  $Dest(s^*, \pi_1(s^*), \pi_2(s^*)) \cap U_q(v_i) \neq \emptyset$ , for some  $q < r$ . Thus we have a contradiction.

It follows from above that there is no closed connected recurrent set of states in  $S \setminus (W_1 \cup T)$ , and hence with probability 1 the game reaches  $W_1 \cup T$  from all states in  $S \setminus (W_1 \cup T)$ . Hence the probability to satisfy  $\text{Safe}(F)$  is equal to the probability to reach  $W_1$ . Since for all states  $s \in S \setminus (W_1 \cup T)$  we have  $Pre_{\pi_1(s), \pi_2(s)}(v_i)(s) \leq v_i(s)$ , it follows that given the strategies  $\pi_1$  and  $\pi_2$ , the valuation  $v_i$  satisfies all the inequalities for linear program to reach  $W_1$ . It follows that the probability to reach  $W_1$  from  $s$  is atmost  $v_i(s)$ . It follows that for all  $s \in S \setminus (W_1 \cup T)$  we have  $\langle\langle 1 \rangle\rangle_{\text{val}}^{\pi_1}(\text{Safe}(F))(s) \leq v_i(s)$ . The result follows. ■

**Convergence.** We first observe that since pure memoryless optimal strategies exist for turn-based stochastic games with safety objectives (Theorem 1), for turn-based stochastic games it suffices to iterate over pure memoryless selectors. Since the number of pure memoryless strategies is finite, it follows for turn-based stochastic games Algorithm 1 always terminates and yields an optimal strategy. For concurrent games, we will use the result that for  $\varepsilon > 0$ , there is a *k-uniform memoryless* strategy that achieves the value of a safety objective with in  $\varepsilon$ . We first define *k-uniform memoryless* strategies. A selector  $\xi$  for player 1 is *k-uniform* if for all  $s \in S \setminus (T \cup W_1)$  and all  $a \in \text{Supp}(\pi_1(s))$  there exists  $i, j \in \mathbb{N}$  such that  $0 \leq i \leq j \leq k$  and  $\xi(s)(a) = \frac{i}{j}$ , i.e., the moves in the support are played with probability that are multiples of  $\frac{1}{k}$  with  $\ell \leq k$ .

**Lemma 4** *For all concurrent game graphs  $G$ , for all safety objectives  $\text{Safe}(F)$ , for  $F \subseteq S$ , for all  $\varepsilon > 0$ , there exist *k-uniform* selectors  $\xi$  such that  $\bar{\xi}$  is an  $\varepsilon$ -optimal strategy for  $k = 2^{\frac{2^{O(n)}}{\varepsilon}}$ , where  $n = |S|$ .*

**Proof.** (*Sketch*). For a rational  $r$ , using the results of [11], it can be shown that whether  $\langle\langle 1 \rangle\rangle_{\text{val}}(\text{Safe}(F))(s) \geq r$  can be expressed in the quantifier free fragment of the theory of reals. Then using the formula in the theory of reals and Theorem 13.12 of [1], it can be shown that if there is a memoryless strategy  $\pi_1$  that achieves value at least  $r$ , then there is a *k-uniform memoryless* strategy  $\pi_1^k$  that achieves value at least  $r - \varepsilon$ , where  $k = 2^{\frac{2^{O(n)}}{\varepsilon}}$ , for  $n = |S|$ . ■

**Strategy improvement with *k-uniform* selectors.** We first argue that if we restrict Algorithm 1 such that every iteration yields a *k-uniform* selector, then the algorithm terminates. If we restrict to *k-uniform* selectors, then a concurrent game graph  $G$  can be converted to a turn-based stochastic game graph, where player 1 first chooses a *k-uniform* selector, then player 2 chooses an action, and then the transition is determined by the chosen *k-uniform* selector of player 1, the action of player 2 and the transition function  $\delta$  of the game graph  $G$ . Then by termination of turn-based stochastic games it follows that the algorithm will terminate. Given  $k$ , let us denote by  $z_i^k$  the valuation of Algorithm 1 at iteration  $i$ , where the selectors are restricted to be *k-uniform*, and  $v_i$  is the valuation of Algorithm 1 at iteration  $i$ . Since  $v_i$  is obtained without any restriction, it follows that for all  $k > 0$ , for all  $i \geq 0$ , we have  $z_i^k \leq v_i$ . From Lemma 4 it follows that for all  $\varepsilon > 0$ , there exists a  $k > 0$  and  $i \geq 0$  such that for all  $s$  we have  $z_i^k(s) \geq \langle\langle 1 \rangle\rangle_{\text{val}}(\text{Safe}(F))(s) - \varepsilon$ . This gives us the following result.

**Theorem 5 (Convergence)** *Let  $v_i$  be the valuation obtained at iteration  $i$  of Algorithm 1. Then the following assertions hold.*

1. *For all  $\varepsilon > 0$ , there exists  $i$  such that for all  $s$  we have  $v_i(s) \geq \langle\langle 1 \rangle\rangle_{\text{val}}(\text{Safe}(F))(s) - \varepsilon$ .*
2.  *$\lim_{i \rightarrow \infty} v_i = \langle\langle 1 \rangle\rangle_{\text{val}}(\text{Safe}(F))$ .*

**Complexity.** Algorithm 1 may not terminate in general. We briefly describe the complexity of every iteration. Given a valuation  $v_i$ , the computation of  $\text{Pre}_1(v_i)$  involves solution of matrix games with rewards  $v_i$  and can be computed in polynomial time using linear-programming. Given  $v_i$  and  $\text{Pre}_1(v_i) = v_i$ , the set  $\text{OptSel}(v_i, s)$  and  $\text{OptSelCount}(v_i, s)$  can be computed by enumerating the subsets of available actions at  $s$  and then using linear-programming: for example to check  $(A, B) \in \text{OptSelCount}(v_i, s)$  it suffices to check that there is an selector  $\xi_1$  such that  $\xi_1$  is optimal (i.e. for all actions  $b \in \Gamma_2(s)$  we have  $\text{Pre}_{\xi_1, b}(v_i)(s) \geq v_i(s)$ ); for all  $a \in A$  we have  $\xi_1(a) > 0$ , and for all  $a \notin A$  we have  $\xi_1(a) = 0$ ; and to check  $B$  is the set of counter-optimal actions we check that for  $b \in B$  we have  $\text{Pre}_{\xi_1, b}(v_i)(s) = v_i(s)$ ; and for  $b \notin B$  we have  $\text{Pre}_{\xi_1, b}(v_i)(s) > v_i(s)$ . All the above can be solved by checking feasibility of a set of linear inequalities. Hence  $\text{TB}(G, v_i, F)$  can be computed in time polynomial in size of  $G$  and  $v_i$  and exponential in the number of moves. The set of almost-sure winning states in turn-based stochastic games with safety objectives can be computed in linear-time [10].

## 5 Termination for Approximation and Turn-based Games

In this section we present termination criteria for strategy improvement algorithms for concurrent games for  $\varepsilon$ -approximation, and then present an improved termination condition for turn-based games.

**Termination for concurrent games.** A strategy improvement algorithm for reachability games was presented in [4]. We refer to the algorithm of [4] as the *reachability strategy improvement algorithm*. The reachability strategy improvement algorithm is simpler than Algorithm 1: it is similar to Algorithm 1 and in every iteration only Step 3.2 is executed (and Step 3.3 need not be executed). Applying the reachability strategy improvement algorithm of [4] for player 2, for a reachability objective  $\text{Reach}(T)$ , we obtain a sequence of valuations  $(u_i)_{i \geq 0}$  such that (a)  $u_{i+1} \geq u_i$ ; (b) if  $u_{i+1} = u_i$ , then  $u_i = \langle\langle 2 \rangle\rangle_{\text{val}}(\text{Reach}(T))$ ; and (c)  $\lim_{i \rightarrow \infty} u_i = \langle\langle 2 \rangle\rangle_{\text{val}}(\text{Reach}(T))$ . Given a concurrent game  $G$  with  $F \subseteq S$  and  $T = S \setminus F$ , we apply the reachability strategy improvement algorithm to obtain the sequence of valuation  $(u_i)_{i \geq 0}$  as above, and we apply Algorithm 1 to obtain a sequence of valuation  $(v_i)_{i \geq 0}$ . The termination criteria are as follows:

1. if for some  $i$  we have  $u_{i+1} = u_i$ , then we have  $u_i = \langle\langle 2 \rangle\rangle_{\text{val}}(\text{Reach}(T))$ , and  $1 - u_i = \langle\langle 1 \rangle\rangle_{\text{val}}(\text{Safe}(F))$ , and we obtain the values of the game;
2. if for some  $i$  we have  $v_{i+1} = v_i$ , then we have  $1 - v_i = \langle\langle 2 \rangle\rangle_{\text{val}}(\text{Reach}(T))$ , and  $v_i = \langle\langle 1 \rangle\rangle_{\text{val}}(\text{Safe}(F))$ , and we obtain the values of the game; and
3. for  $\varepsilon > 0$ , if for some  $i \geq 0$ , we have  $u_i + v_i \geq 1 - \varepsilon$ , then for all  $s \in S$  we have  $v_i(s) \geq \langle\langle 1 \rangle\rangle_{\text{val}}(\text{Safe}(F))(s) - \varepsilon$  and  $u_i(s) \geq \langle\langle 2 \rangle\rangle_{\text{val}}(\text{Reach}(T))(s) - \varepsilon$  (i.e., the algorithm can stop for  $\varepsilon$ -approximation).

Observe that since  $(u_i)_{i \geq 0}$  and  $(v_i)_{i \geq 0}$  are both monotonically non-decreasing and  $\langle\langle 1 \rangle\rangle_{\text{val}}(\text{Safe}(F)) + \langle\langle 2 \rangle\rangle_{\text{val}}(\text{Reach}(T)) = 1$ , it follows that if  $u_i + v_i \geq 1 - \varepsilon$ , then for all  $j \geq i$  we have  $u_i \geq u_j - \varepsilon$  and  $v_i \geq v_j - \varepsilon$ . This establishes that  $u_i \geq \langle\langle 1 \rangle\rangle_{\text{val}}(\text{Safe}(F)) - \varepsilon$  and  $v_i \geq \langle\langle 2 \rangle\rangle_{\text{val}}(\text{Reach}(T)) - \varepsilon$ ; and the correctness of the stopping criteria (3) for  $\varepsilon$ -approximation follows. We also note that instead of applying the reachability strategy improvement algorithm, a value-iteration algorithm can be applied for reachability games to obtain a sequence of valuation with properties similar to  $(u_i)_{i \geq 0}$  and the above termination criteria can be applied.

**Theorem 6** *Let  $G$  be a concurrent game graph with a safety objective  $\text{Safe}(F)$ . Algorithm 1 and the reachability strategy improvement algorithm for player 2 for the reachability objective  $\text{Reach}(S \setminus F)$  yield sequence of valuations  $(v_i)_{i \geq 0}$  and  $(u_i)_{i \geq 0}$ , respectively, such that (a) for all  $i \geq 0$ , we have  $v_i \leq \langle\langle 1 \rangle\rangle_{\text{val}}(\text{Safe}(F)) \leq 1 - u_i$ ; and (b)  $\lim_{i \rightarrow \infty} v_i = \lim_{i \rightarrow \infty} 1 - u_i = \langle\langle 1 \rangle\rangle_{\text{val}}(\text{Safe}(F))$ .*

**Termination for turn-based games.** For turn-based stochastic games Algorithm 1 and as well as the reachability strategy improvement algorithm terminates. Each iteration of the reachability strategy improvement algorithm of [4] is computable in polynomial time, and here we present a termination guarantee for the reachability strategy improvement algorithm. To apply the reachability strategy improvement algorithm we assume the objective of player 1 to be a reachability objective  $\text{Reach}(T)$ , and the correctness of the algorithm relies on the notion of *proper strategies*. Let  $W_2 = \{s \in S \mid \langle\langle 1 \rangle\rangle_{\text{val}}(\text{Reach}(T))(s) = 0\}$ . Then the notion of proper strategies and its properties are as follows.

**Definition 4 (Proper strategies and selectors)** *A player-1 strategy  $\pi_1$  is proper if for all player-2 strategies  $\pi_2$ , and for all states  $s \in S \setminus (T \cup W_2)$ , we have  $\Pr_s^{\pi_1, \pi_2}(\text{Reach}(T \cup W_2)) = 1$ . A player-1 selector  $\xi_1$  is proper if the memoryless player-1 strategy  $\bar{\xi}_1$  is proper.*

**Lemma 5 ([4])** *Given a selector  $\xi_1$  for player 1, the memoryless player-1 strategy  $\bar{\xi}_1$  is proper iff for every pure selector  $\xi_2$  for player 2, and for all states  $s \in S$ , we have  $\Pr_s^{\bar{\xi}_1, \xi_2}(\text{Reach}(T \cup W_2)) = 1$ .*

The following result follows from the result of [4] specialized for the case of turn-based stochastic games.

**Lemma 6** *Let  $G$  be a turn-based stochastic game with reachability objective  $\text{Reach}(T)$  for player 1. Let  $\gamma_0$  be the initial selector, and  $\gamma_i$  be the selector obtained at iteration  $i$  of the reachability strategy improvement algorithm. If  $\gamma_i$  is a pure, proper selector, then the following assertions hold:*

1. *for all  $i \geq 0$ , we have  $\gamma_i$  is a pure, proper selector;*
2. *for all  $i \geq 0$ , we have  $u_{i+1} \geq u_i$ , where  $u_i = \langle\langle 1 \rangle\rangle_{\text{val}}^{\gamma_i}(\text{Reach}(T))$  and  $u_{i+1} = \langle\langle 1 \rangle\rangle_{\text{val}}^{\gamma_{i+1}}(\text{Reach}(T))$ ; and*
3. *if  $u_{i+1} = u_i$ , then  $u_i = \langle\langle 1 \rangle\rangle_{\text{val}}(\text{Reach}(T))$ , and there exists  $i$  such that  $u_{i+1} = u_i$ .*

The strategy improvement algorithm of Condon [6] works only for *halting games*, but the reachability strategy improvement algorithm works if we start with a pure, proper selector for reachability games that are not halting. Hence to use the reachability strategy improvement algorithm to compute values we need to start with a pure, proper selector. We present a procedure to compute a



pure, proper selector, and then present termination bounds (i.e., bounds on  $i$  such that  $u_{i+1} = u_i$ ). The construction of pure, proper selector is based on the notion of *attractors* defined below.

*Attractor strategy.* Let  $A_0 = W_2 \cup T$ , and for  $i \geq 0$  we have

$$A_{i+1} = A_i \cup \{s \in S_1 \cup S_R \mid E(s) \cap A_i \neq \emptyset\} \cup \{s \in S_2 \mid E(s) \subseteq A_i\}.$$

Since for all  $s \in S \setminus W_2$  we have  $\langle\langle 1 \rangle\rangle_{\text{val}}(\text{Reach}(T)) > 0$ , it follows that from all states in  $S \setminus W_2$  player 1 can ensure that  $T$  is reached with positive probability. It follows that for some  $i \geq 0$  we have  $A_i = S$ . The pure *attractor* selector  $\xi^*$  is as follows: for a state  $s \in (A_{i+1} \setminus A_i) \cap S_1$  we have  $\xi^*(s)(t) = 1$ , where  $t \in A_i$  (such a  $t$  exists by construction). The pure memoryless strategy  $\bar{\xi}^*$  ensures that for all  $i \geq 0$ , from  $A_{i+1}$  the game reaches  $A_i$  with positive probability. Hence there is no end-component  $C$  contained in  $S \setminus (W_2 \cup T)$  in the MDP  $G_{\bar{\xi}^*}$ . It follows that  $\xi^*$  is a pure selector that is proper, and the selector  $\xi^*$  can be computed in  $O(|E|)$  time. This completes the reachability strategy improvement algorithm for turn-based stochastic games. We now present the termination bounds.

*Termination bounds.* We present termination bounds for binary turn-based stochastic games. A turn-based stochastic game is binary if for all  $s \in S_R$  we have  $|E(s)| \leq 2$ , and for all  $s \in S_R$  if  $|E(s)| = 2$ , then for all  $t \in E(s)$  we have  $\delta(s)(t) = \frac{1}{2}$ , i.e., for all probabilistic states there are at most two successors and the transition function  $\delta$  is uniform.

**Lemma 7** *Let  $G$  be a binary Markov chain with  $|S|$  states with a reachability objective  $\text{Reach}(T)$ . Then for all  $s \in S$  we have  $\langle\langle 1 \rangle\rangle_{\text{val}}(\text{Reach}(T)) = \frac{p}{q}$ , with  $p, q \in \mathbb{N}$  and  $p, q \leq 4^{|S|-1}$ .*

**Proof.** The results follow as a special case of Lemma 2 of [6]. Lemma 2 of [6] holds for halting turn-based stochastic games, and since Markov chains reaches the set of closed connected recurrent states with probability 1 from all states the result follows. ■

**Lemma 8** *Let  $G$  be a binary turn-based stochastic game with a reachability objective  $\text{Reach}(T)$ . Then for all  $s \in S$  we have  $\langle\langle 1 \rangle\rangle_{\text{val}}(\text{Reach}(T)) = \frac{p}{q}$ , with  $p, q \in \mathbb{N}$  and  $p, q \leq 4^{|S_R|-1}$ .*

**Proof.** Since pure memoryless optimal strategies exist for both players (Theorem 1), we fix pure memoryless optimal strategies  $\pi_1$  and  $\pi_2$  for both players. The Markov chain  $G_{\pi_1, \pi_2}$  can be then reduced to an equivalent Markov chains with  $|S_R|$  states (since we fix deterministic successors for states in  $S_1 \cup S_2$ , they can be collapsed to their successors). The result then follows from Lemma 7. ■

From Lemma 8 it follows that at iteration  $i$  of the reachability strategy improvement algorithm either the sum of the values either increases by  $\frac{1}{4^{|S_R|-1}}$  or else there is a valuation  $u_i$  such that  $u_{i+1} = u_i$ . Since the sum of values of all states can be at most  $|S|$ , it follows that algorithm terminates in at most  $|S| \cdot 4^{|S_R|-1}$  steps. Moreover, since the number of pure memoryless strategies is at most  $\prod_{s \in S_1} |E(s)|$ , the algorithm terminates in at most  $\prod_{s \in S_1} |E(s)|$  steps. It follows from the results of [19] that a turn-based stochastic game graph  $G$  can be reduced to a equivalent binary turn-based stochastic game graph  $G'$  such that the set of player 1 and player 2 states in  $G$  and  $G'$  are the same and the number of probabilistic states in  $G'$  is  $O(|\delta|)$ , where  $|\delta|$  is the size of the transition function in  $G$ . Thus we obtain the following result.

**Theorem 7** *Let  $G$  be a turn-based stochastic game with a reachability objective  $\text{Reach}(T)$ , then the reachability strategy improvement algorithm computes the values in time*

$$O\left(\min\left\{\prod_{s \in S_1} |E(s)|, 2^{O(|\delta|)}\right\} \cdot \text{poly}(|G|)\right);$$

where  $\text{poly}$  is polynomial function.

The results of [15] presented an algorithm for turn-based stochastic games that works in time  $O(|S_R|! \cdot \text{poly}(|G|))$ . The algorithm of [15] works only for turn-based stochastic games, for general turn-based stochastic games the complexity of the algorithm of [15] is better. However, for turn-based stochastic games where the transition function at all states can be expressed in constant bits we have  $|\delta| = O(|S_R|)$ . In these cases the reachability strategy improvement algorithm (that works for both concurrent and turn-based stochastic games) works in time  $2^{O(|S_R|)} \cdot \text{poly}(|G|)$  as compared to the time  $2^{O(|S_R| \cdot \log(|S_R|))} \cdot \text{poly}(|G|)$  of the algorithm of [15].

## References

- [1] S. Basu, R. Pollack, and M.-F. Roy. *Algorithms in Real Algebraic Geometry*. Springer, 2003.
- [2] D.P. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific, 1995. Volumes I and II.
- [3] A. Bianco and L. de Alfaro. Model checking of probabilistic and nondeterministic systems. In *FSTTCS 95: Software Technology and Theoretical Computer Science*, volume 1026 of *Lecture Notes in Computer Science*, pages 499–513. Springer-Verlag, 1995.
- [4] K. Chatterjee, L. de Alfaro, and T.A. Henzinger. Strategy improvement in concurrent reachability games. In *QEST'06*. IEEE, 2006.
- [5] A. Condon. The complexity of stochastic games. *Information and Computation*, 96(2):203–224, 1992.
- [6] A. Condon. On algorithms for simple stochastic games. In *Advances in Computational Complexity Theory*, volume 13 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 51–73. American Mathematical Society, 1993.
- [7] C. Courcoubetis and M. Yannakakis. The complexity of probabilistic verification. *Journal of the ACM*, 42(4):857–907, 1995.
- [8] L. de Alfaro. *Formal Verification of Probabilistic Systems*. PhD thesis, Stanford University, 1997. Technical Report STAN-CS-TR-98-1601.
- [9] L. de Alfaro and T.A. Henzinger. Concurrent omega-regular games. In *Proceedings of the 15th Annual Symposium on Logic in Computer Science*, pages 141–154. IEEE Computer Society Press, 2000.
- [10] L. de Alfaro, T.A. Henzinger, and O. Kupferman. Concurrent reachability games. *Theoretical Computer Science*, 386(3):188–217, 2007.

- [11] L. de Alfaro and R. Majumdar. Quantitative solution of omega-regular games. *Journal of Computer and System Sciences*, 68:374–397, 2004.
- [12] C. Derman. *Finite State Markovian Decision Processes*. Academic Press, 1970.
- [13] K. Etessami and M. Yannakakis. Recursive concurrent stochastic games. In *ICALP 06: Automata, Languages, and Programming*. Springer, 2006.
- [14] J. Filar and K. Vrieze. *Competitive Markov Decision Processes*. Springer-Verlag, 1997.
- [15] H. Gimbert and F. Horn. Simple stochastic games with few random vertices are easy to solve. In *FoSSaCS'08 (To appear)*, 2008.
- [16] J.G. Kemeny, J.L. Snell, and A.W. Knapp. *Denumerable Markov Chains*. D. Van Nostrand Company, 1966.
- [17] P.R. Kumar and T.H. Shiau. Existence of value and randomized strategies in zero-sum discrete-time stochastic dynamic games. *SIAM J. Control and Optimization*, 19(5):617–634, 1981.
- [18] D.A. Martin. The determinacy of Blackwell games. *The Journal of Symbolic Logic*, 63(4):1565–1581, 1998.
- [19] U. Zwick and M.S. Paterson. The complexity of mean payoff games on graphs. *Theoretical Computer Science*, 158:343–359, 1996.